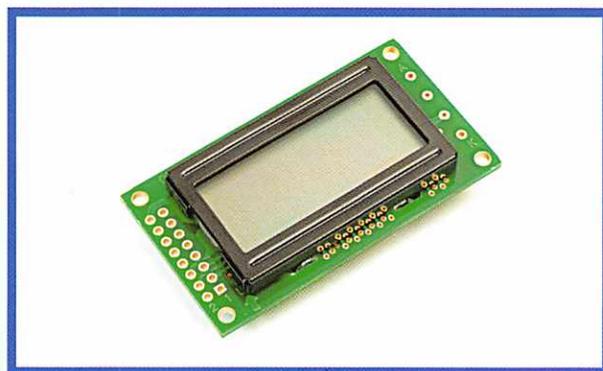
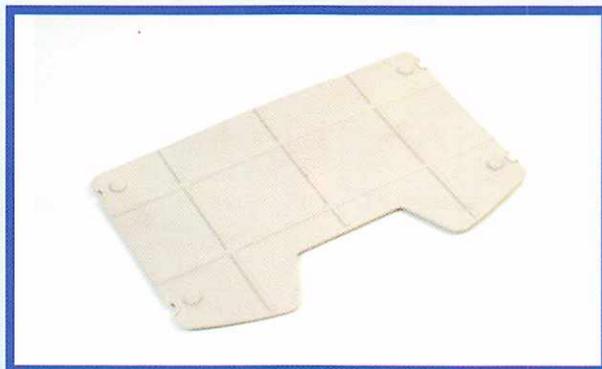
Direttore responsabile:
ALBERTO PERUZZO
Direttore Grandi Opere:
GIORGIO VERCELLINI
Consulenza tecnica e traduzioni:
CONSULCOMP S.n.c.
Pianificazione tecnica
LEONARDO PITTON

Direzione, Redazione, Amministrazione: viale Ercole Marelli 165, Tel. 02/242021, 20099 Sesto San Giovanni (Mi). Pubblicazione settimanale. Registrazione del Tribunale di Monza n. 1738 del 26/05/2004. Spedizione in abbonamento postale gr. II/70; autorizzazione delle Poste di Milano n. 163464 del 13/2/1963. Stampa: Grafiche Porpora s.r.l., Cernusco S/N (MI). Distribuzione SO.DI.P. S.p.A., Cinisello Balsamo (MI).

"ELETTRONICA DIGITALE" si compone di 70 fascicoli settimanali da suddividere in 2 raccoglitori.

RICHIESTA DI NUMERI ARRETRATI. Per ulteriori informazioni, telefonare dal lunedì al venerdì ore 9.30-12.30 all'ufficio arretrati tel. 02/242021. Se vi mancano dei fascicoli o dei raccoglitori per completare l'opera, e non li trovate presso il vostro edicolante, potrete riceverli a domicilio rivolgendovi direttamente alla casa editrice. Basterà compilare e spedire un bollettino di conto corrente postale a PERUZZO & C. s.r.l., Ufficio Arretrati, viale Marelli 165, 20099 Sesto San Giovanni (MI). Il nostro numero di c/c postale è 42980201. L'importo da versare sarà pari al prezzo dei fascicoli o dei raccoglitori richiesti, più le spese di spedizione € 3,10 per pacco. Qualora il numero dei fascicoli o dei raccoglitori sia tale da superare il prezzo globale di € 25,82 e non superiore a € 51,65, l'invio avverrà per pacco assicurato e le spese di spedizione ammonteranno a € 6,20. La spesa sarà di € 9,81 da € 51,65 a € 103,29; di € 12,39 da € 103,29 a € 154,94; di € 14,98 da € 154,94 a € 206,58; di € 16,53 da € 206,58 in su. Attenzione: ai fascicoli arretrati, trascorse dodici settimane dalla loro distribuzione in edicola, viene applicato un sovrapprezzo di € 0,52, che andrà pertanto aggiunto all'importo da pagare. Non vengono effettuate spedizioni contrassegno. Gli arretrati di fascicoli e raccoglitori saranno disponibili per un anno dal completamento dell'opera. IMPORTANTE: è assolutamente necessario specificare sul bollettino di c/c postale, nello spazio riservato alla causale del versamento, il titolo dell'opera nonché il numero dei fascicoli e dei raccoglitori che volete ricevere.

IN REGALO in questo fascicolo

Display LCD
da 2 file x 8 caratteri

IN REGALO nel prossimo fascicolo

1 Pannello di chiusura per la parte inferiore del laboratorio

COME RACCOGLIERE E SUDDIVIDERE L'OPERA NELLE 4 SEZIONI

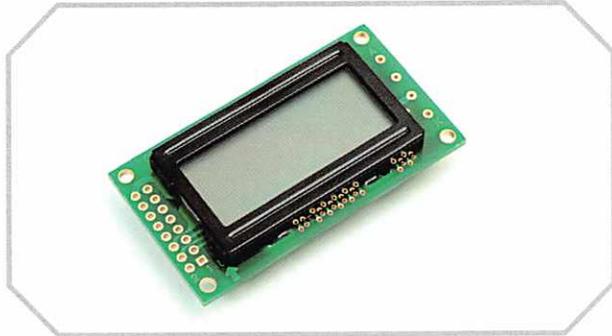
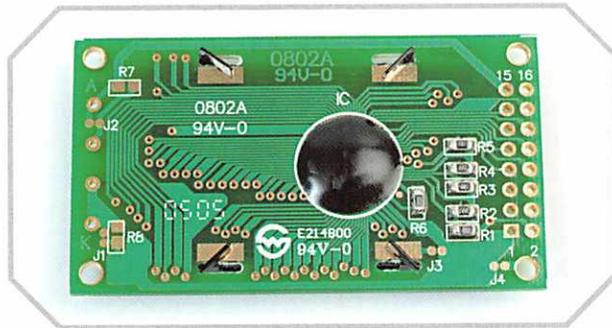
L'Opera è composta da 4 sezioni identificabili dalle fasce colorate, come indicato sotto. Le schede di ciascun fascicolo andranno suddivise nelle sezioni indicate e raccolte nell'apposito raccoglitore, che troverai presto in edicola. Per il momento, ti consigliamo di suddividere le sezioni in altrettante cartellette, in attesa di poterle collocare nel raccoglitore. A prima vista, alcuni numeri di pagina ti potranno sembrare ripetuti o sbagliati. Non è così: ciascuno fa parte di sezioni differenti e rispecchia l'ordine secondo cui raccogliere le schede. **Per eventuali domande di tipo tecnico scrivere al seguente indirizzo e-mail:** elettronicadigitale@microrobots.it

Hardware Montaggio e prove del laboratorio

Digitale di base Esercizi con i circuiti digitali

Digitale avanzato Esercizi con i circuiti sequenziali

Microcontroller Esercizi con i microcontroller

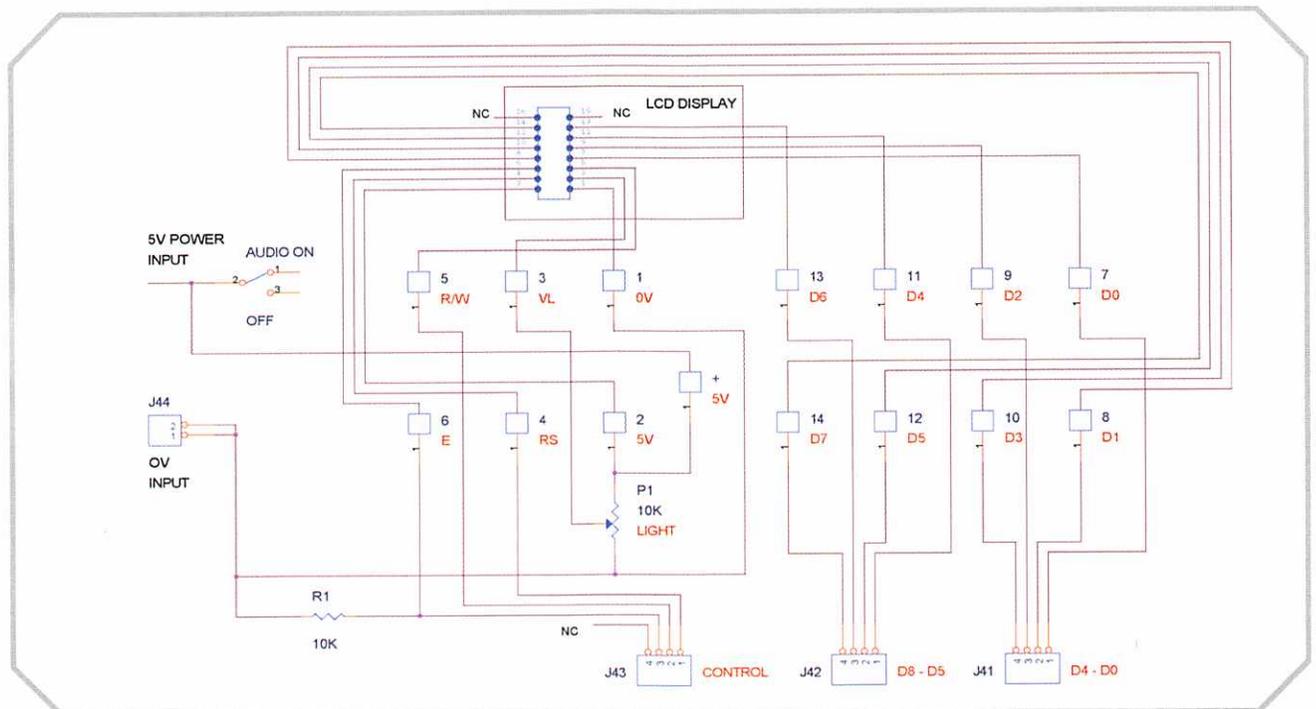
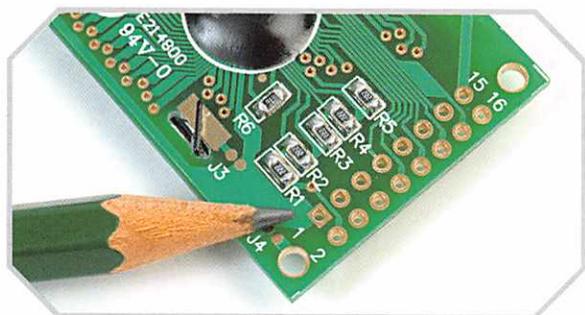
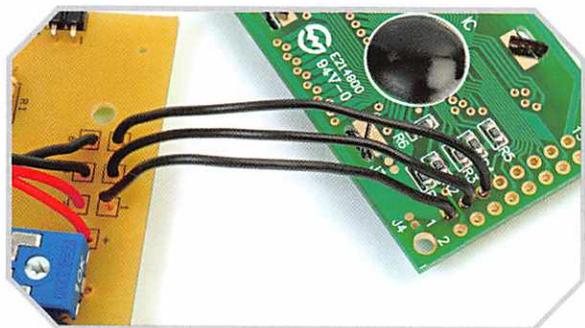
Il display LCD

Display.

Vista posteriore del display.

Con questo fascicolo viene fornito il dispositivo di visualizzazione LCD, che si può anche chiamare display LCD.

Installazione

Il lavoro necessario per installare il display si divide in diverse operazioni. La prima consiste nell'eseguire tutti i collegamenti tra questa scheda, che contiene il display, e la DG14, che contiene i collegamenti esterni e quello di alimentazione. La seconda consiste nel collegamento del positivo dell'alimentazione e la terza comprende l'installazione di entrambe le schede, fissandole con le viti fornite in precedenza.

Collegamenti fra le schede

I collegamenti fra le schede si eseguono con cablaggi diretti. Questa operazione può risultare piuttosto veloce se è già stato eseguito il lavoro indicato nel fascicolo precedente, che

Schema elettrico.

I collegamenti sono posizionati su due file.

Primi collegamenti a 1, 3 e 5.

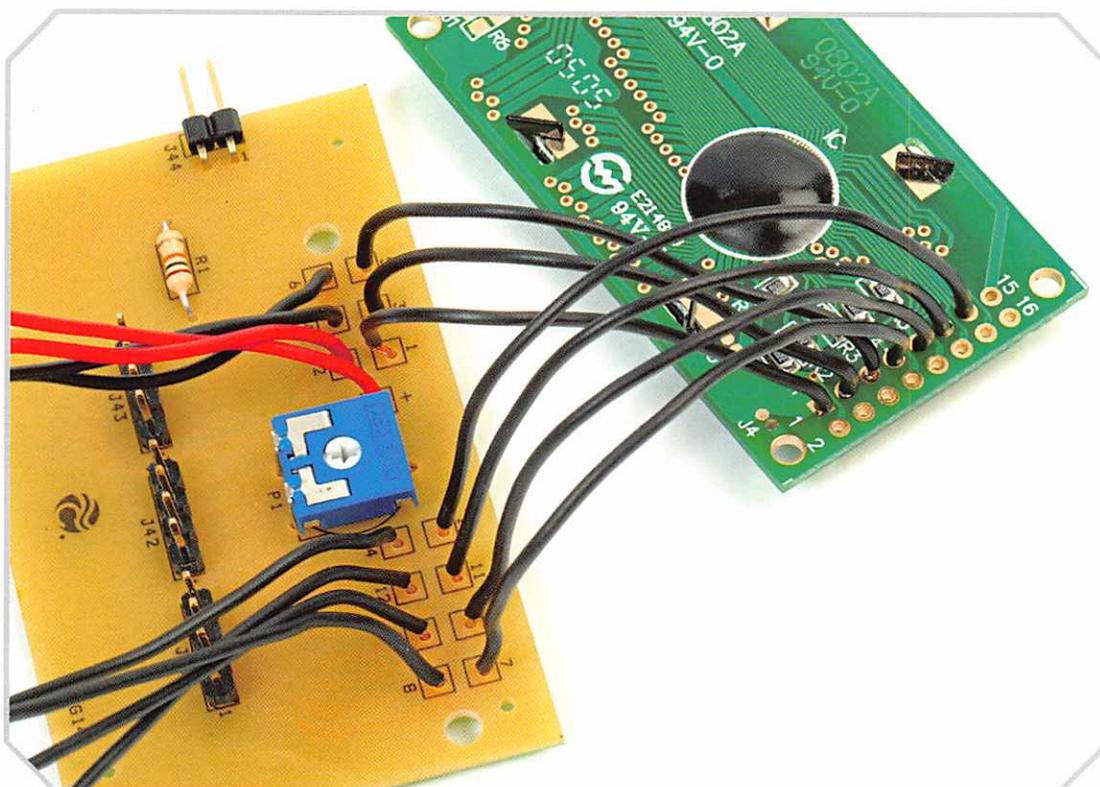
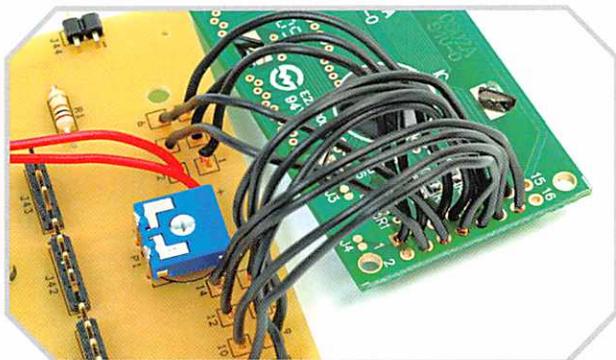
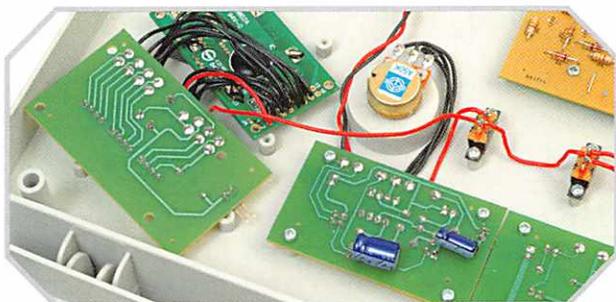
comprende il collegamento di tutti i fili necessari dal lato della scheda DG14.

Prima di iniziare il cablaggio è necessario osservare bene la scheda del display, troveremo due file di piazzole di collegamento, una per i collegamenti pari dal 2 al 16 e l'altra per i collegamenti dispari dall'1 al 15, quest'ultima situata verso l'interno della scheda.

Le due schede devono essere posizionate come indicato dalle fotografie, fuori dal laboratorio per facilitare il lavoro e per evitare di danneggiarlo accidentalmente.

Cablaggio

I collegamenti si eseguono in modo che le piazzole di collegamento di una e dell'altra scheda abbiano lo stesso riferimento. Cominceremo dal collegamento 1 e continueremo con 3, 5, 7, 9, 11 e 13. Dato che la piazzola 15 della scheda e del display non è utilizzata, la lasceremo libera.

Proseguiremo con i collegamenti pari, cioè con le connessioni 14, 12, 10 e 8. Prima di continuare con i collegamenti corrispondenti alle piazzole 4 e 6 osservate bene le fotografie. In-

Display con le connessioni dispari.

Connessioni pari.

Collegamento del positivo dell'alimentazione.

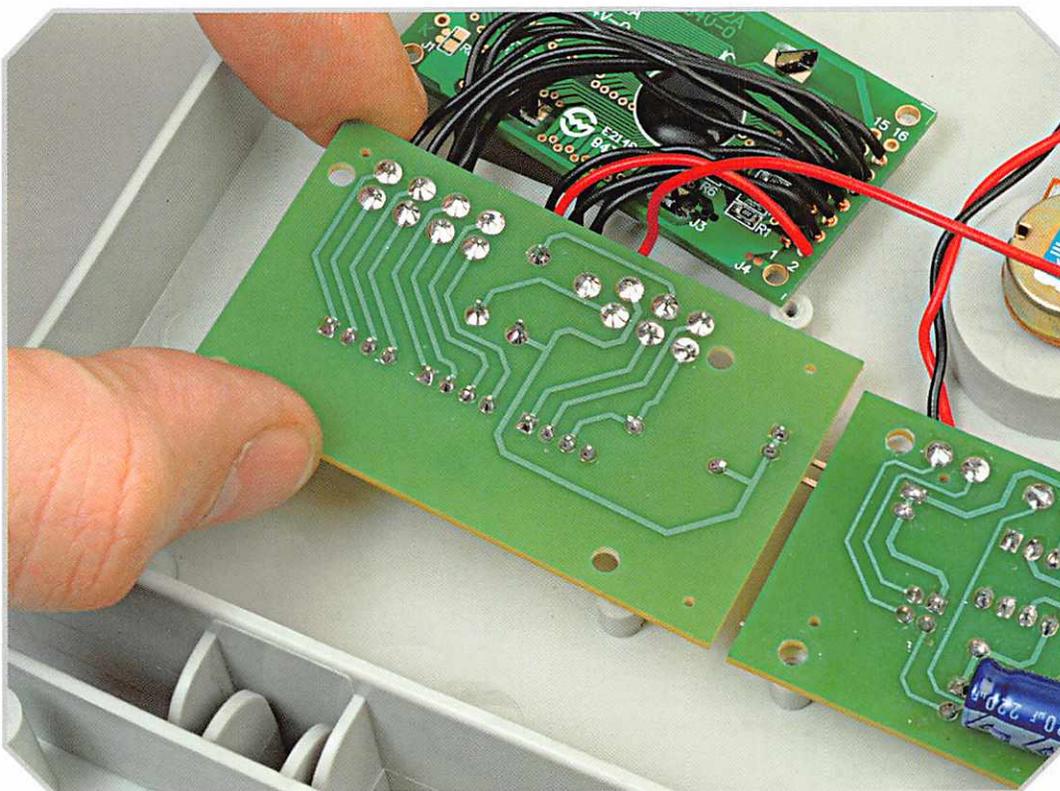
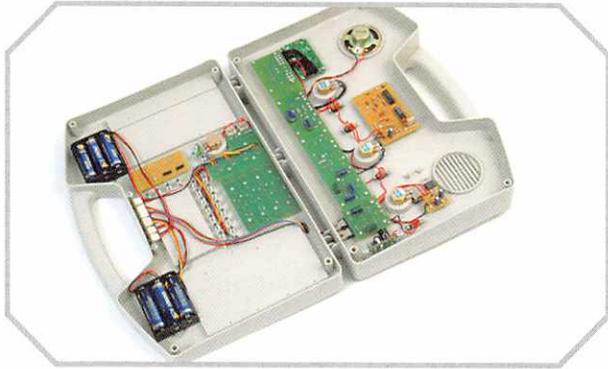
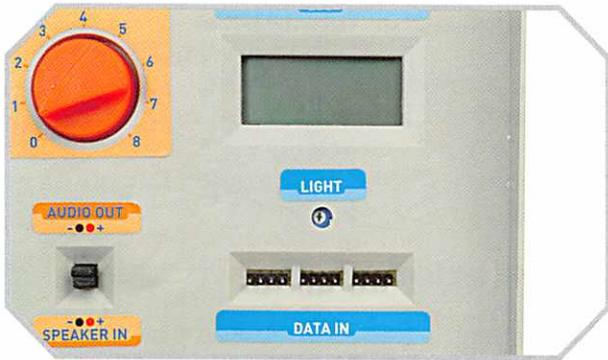
fine rimarrà solamente la piazzola 2 che corrisponde al positivo dell'alimentazione e con la quale concluderemo il collegamento delle schede.

Collegamento dell'alimentazione

Rimane solamente da eseguire una saldatura, quella del collegamento del positivo dell'alimentazione, prima però è necessario collocare le schede in una posizione molto vicina alla definitiva. Il cursore del potenziometro deve essere situato all'incirca a metà della sua corsa.

Il collegato al terminale + della scheda DG14 si collega alla linea di alimentazione da 5 V e si prende dal terminale centrale dell'ultimo commutatore di alimentazione, cioè quello AUDIO (ON/OFF). Il negativo dell'alimentazione si collega inserendo il connettore J44 nella scheda adiacente.

Installazione delle schede

Dopo che le schede sono state portate molto vicino alla loro posizione definitiva, si posizio-

Collegamento della scheda DG14 alla DG15.

Vista dell'interno del laboratorio.

Zona del display e relativi collegamenti.

nano correttamente i fili, al loro posto senza forzarli.

Per poter inserire la scheda DG14 è necessario togliere le quattro viti che fissano la scheda audio DG15, allo scopo di poterla sollevare e inserire il connettore corrispondente al negativo dell'alimentazione, dopodiché si collocheranno entrambe le schede nelle loro posizioni.

Ora è possibile collegare dal lato del pannello frontale i cavetti con i connettori a 4 vie su J41 e J43 e i ponticelli sulla scheda audio, in modo da centrare bene i connettori; quindi passeremo all'inserimento delle 8 viti, 4 per ogni scheda e si stringeranno in modo progressivo.

Potrebbe succedere che la scheda DG14 rimanga un po' forzata, a causa dell'altezza dei potenziometri.

Il passo successivo sarà fissare la scheda del display, utilizzando le quattro viti che verranno chiuse progressivamente, in pratica le inseriremo dando solamente un giro e poi le chiuderemo in diagonale, stringendo a turno le viti sino a finecorsa, tenendo presente che devono solo fissare il display e che non è necessario forzarle.

Vista generale del laboratorio.

VCO audio con transistor

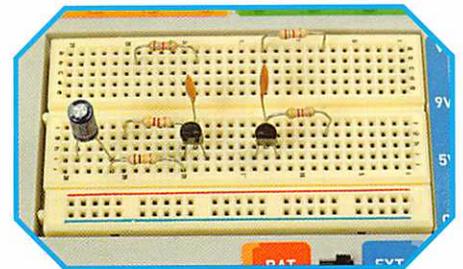
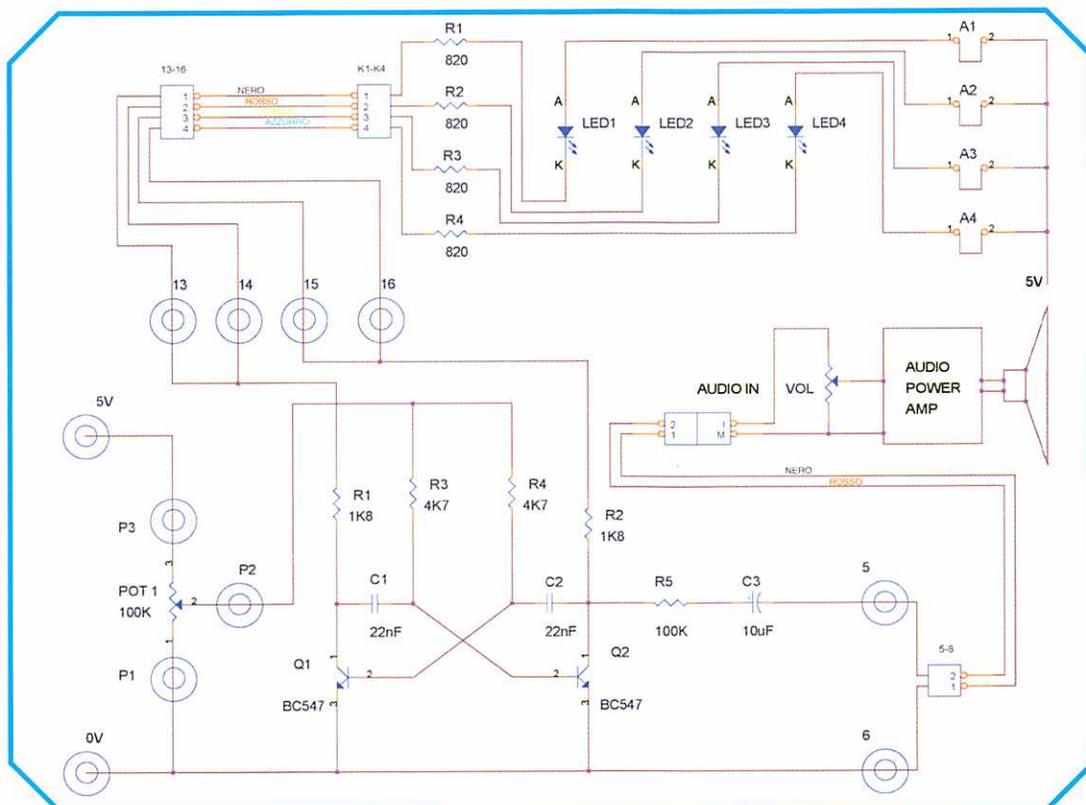
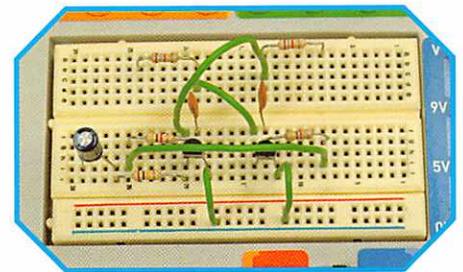
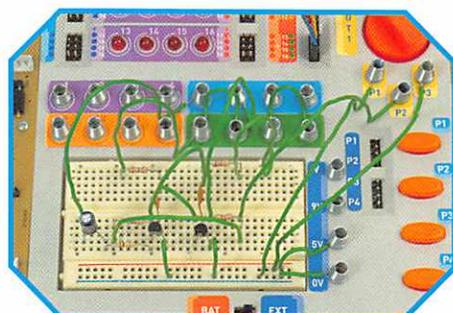
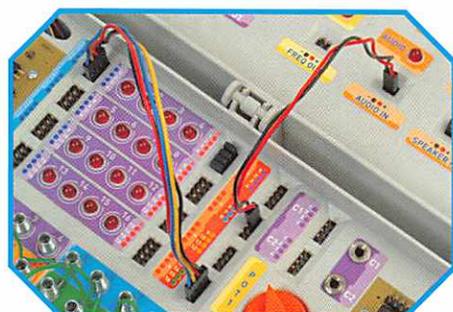
***Q**uesto circuito corrisponde alla configurazione classica di un VCO (Oscillatore Controllato in Voltaggio), è costruito con soli due transistor NPN ed è progettato in modo da lavorare nella banda audio.*

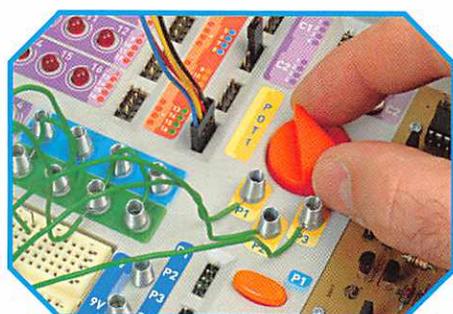
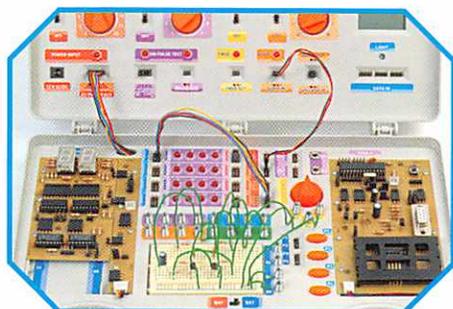
Il circuito

Osservando il circuito vedremo che ci risulterà familiare, benché in questo caso lavori con frequenze comprese all'interno della banda audio, in cui l'udito può apprezzare chiaramente le variazioni di frequenza, infatti nell'essere umano questo senso è molto sviluppato.

La frequenza di oscillazione dipende dai valori delle resistenze R3 e R4, dai condensatori C1 e C2 e dalla tensione presente sul punto di collegamento delle resistenze R3, R4; questa tensione è a sua volta determinata dalla posizione del cursore del potenziometro POT 1. Questa tensione può essere regolata fra 0 e 5 V.

L'uscita del circuito si attenua con la resistenza R5, inoltre la presenza del condensatore C3 garantisce che esca solamente la componente alternata, che contiene realmente l'in-

Componenti sulla scheda Bread Board.

Collegamenti della scheda.

Schema elettrico.

Dettaglio del collegamento alle molle.

Collegamento all'amplificatore audio.

Il commutatore audio deve essere su ON.

Ruotando POT 1 cambia la frequenza.

Esperimento completato.

formazione audio. Il segnale elettrico generato si applica all'amplificatore audio in modo che possa essere ascoltato sull'altoparlante del laboratorio.

I LED 1 e 2 e i LED 3 e 4 si illuminano alternativamente, benché noi li vediamo costantemente illuminati, infatti l'occhio umano non è in grado di percepire questa alternanza; quindi questo circuito serve solamente per verificare che circoli corrente sui circuiti di collettori di entrambi i transistor.

Montaggio

È necessario fare attenzione all'inserimento dei transistor sulla scheda Bread Board, e anche ai condensatori elettrolitici. Dobbiamo inserire i quattro ponticelli sugli anodi dei diodi da LED 1 a LED 4 e collegare un cavetto tra i collegamenti dei catodi di questi diodi e i connettori corrispondenti alle molle dalla 13 alla 16. I collegamenti dalla scheda Bread Board all'amplificatore audio si eseguono con un cavetto terminato su due connettori a due vie, uno dei quali si porta al connettore corrispondente alle molle 5 e 6 e l'altro capo al connettore AUDIO IN, tenendo presente che il filo rosso corrisponde al punto rosso, e quello nero al punto nero. È necessario inserire anche i due ponticelli di collegamento in senso orizzontale, fra l'uscita dell'amplificatore audio e l'altoparlante.

Prova

Se tutti i componenti sono stati correttamente montati e tutti i collegamenti eseguiti, il circuito deve funzionare al primo tentativo. Dobbiamo alimentare l'amplificatore audio con il commutatore AUDIO ON. Il comando del potenziometro POT 1 deve essere posizionato all'incirca a metà della sua corsa. Dopo aver avviato il circuito si udirà un suono sull'altoparlante, la cui frequenza si può modificare azionando il comando POT 1. È necessario tener presente che con tensioni molto basse l'oscillatore cessa di funzionare.

LISTA DEI COMPONENTI

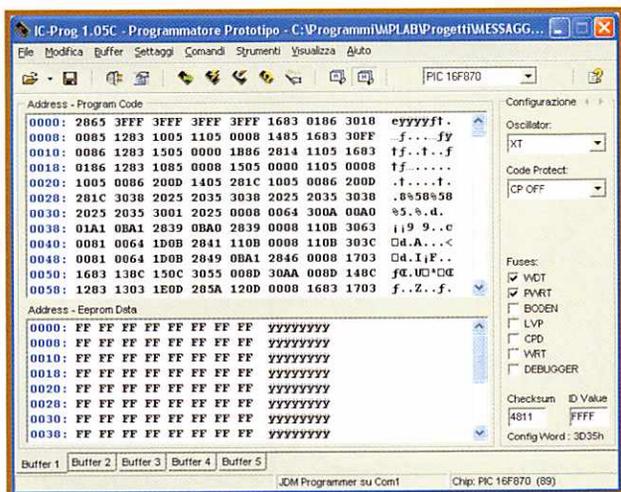
Q1, Q2	Transistor BC547 o BC548
R1, R2	Resistenza 1K8 (marrone, grigio, rosso)
R3, R4	Resistenza 4K7 (giallo, viola, rosso)
R5	Resistenza 100 K (marrone, nero, giallo)
C1, C2	Condensatore 22 nF
C3	Condensatore 10 µF elettrolitico

Esercizio: generatore di messaggi, la pratica

Concludiamo l'esercizio del generatore di messaggi scrivendo il programma sul microcontroller e realizzando il montaggio hardware dell'applicazione. Benché le applicazioni che comprendono il display LCD richiedano più programmazione o una programmazione più complessa rispetto ad altri tipi di applicazioni, i montaggi dei circuiti risultano piuttosto semplici.

Scrittura del PIC

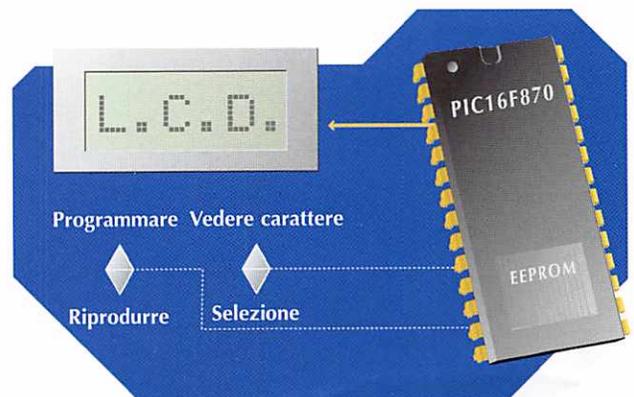
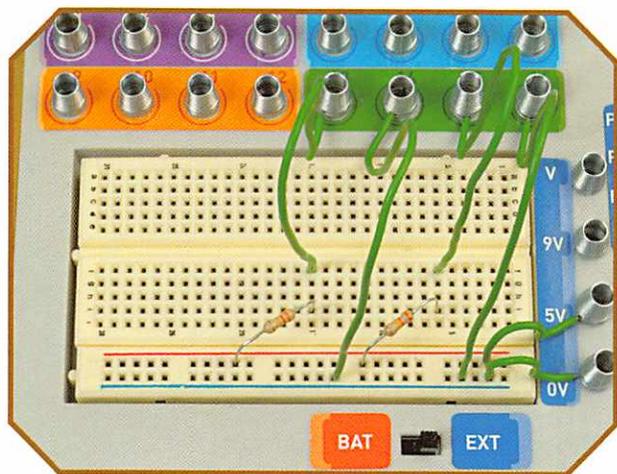
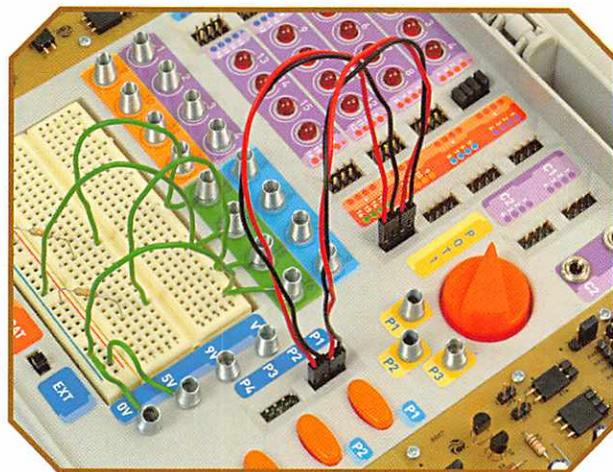
Dopo aver simulato il programma e verificato che non abbia errori, lo scriveremo sul PIC. Ricordate che per la scrittura è necessario predisporre il laboratorio, a questo scopo collegheremo il cavo di trasferimento e inseriremo i ponticelli sui connettori JP8 e JP9 e sui connettori JP1, JP2 e JP3 sulle posizioni 1 e 2. Non dimenticate che i terminali della porta B devono essere scollegati, in quanto potrebbero generare errori nel processo di scrittura.

Facciamo partire IC-Prog ed eseguiamo le fasi di preparazione alla scrittura (lettura, cancellazione e verifica). Apriamo il file che vogliamo trasferire sul PIC, "messaggio.asm", e configuriamo IC-Prog in modo che il trasferimento venga eseguito correttamente. Per questo cambieremo il tipo di oscillatore a XT, verificheremo che la protezione del codice sia CP OFF e selezioneremo WDT e PWRT tra i bit di configurazione. A questo punto daremo il via alla scrittura selezionando l'opzione Programma Tutto. Terminata la fase di scrittura verificheremo il processo eseguendo una lettura del contenuto del PIC.

Carichiamo il programma su IC-Prog.

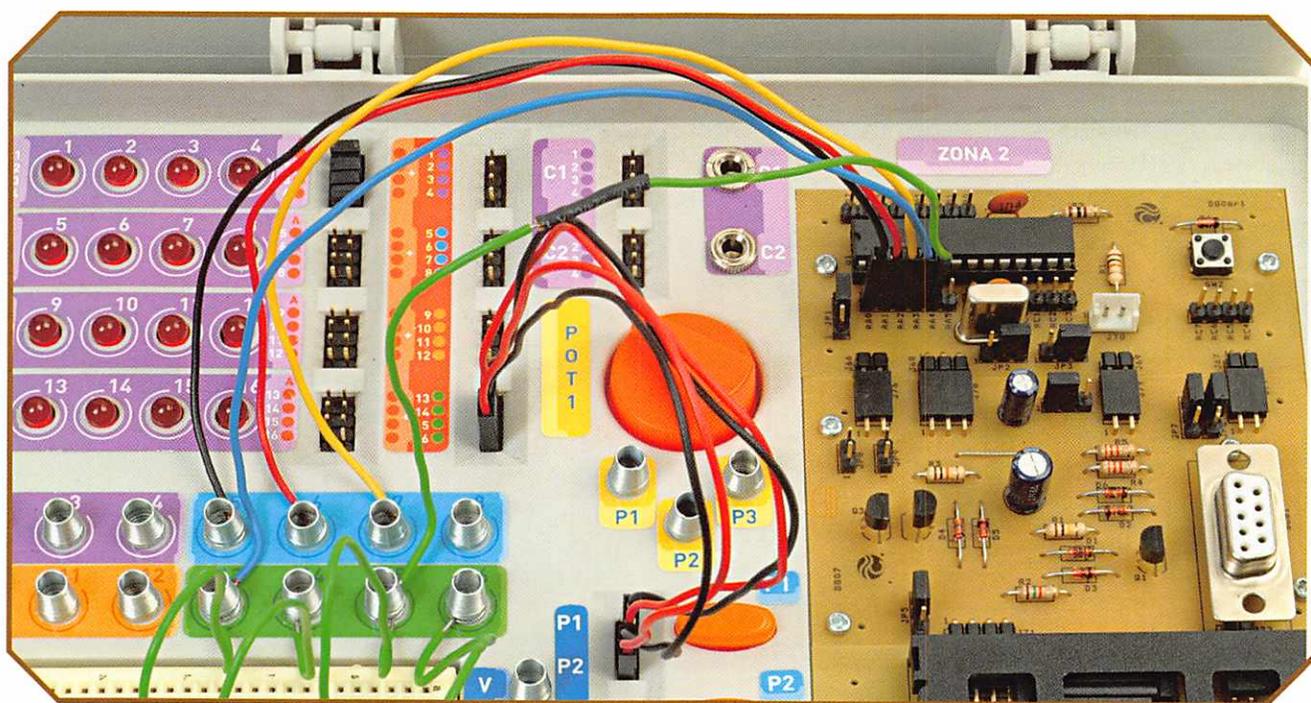
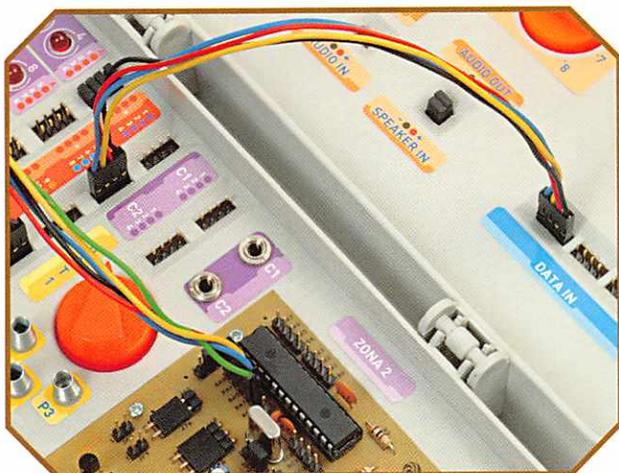
Montaggio hardware

Facendo riferimento allo schema generale di funzionamento separeremo il montaggio in due parti: montaggio degli ingressi (RA3 e RA4) e montaggio delle uscite (RA2:RA0 e la porta B). Prima di procedere ai montaggi dobbiamo riportare il laboratorio in modo funzionamento, quindi scollegheremo il cavo di trasferimento dei dati tra il PC e il laboratorio, toglieremo i ponticelli da JP8 e JP9 e sposteremo i ponticelli di JP1, JP2 e JP3 dalla loro posizione, portandoli sui pin 2 e 3.

Montaggio degli ingressi

Abbiamo bisogno di due ingressi, uno per selezionare programmare/riprodurre e l'altro per selezione/vedi caratteri, i quali all'interno del codice sono stati assegnati a RA4 e RA3 rispettivamente. Nella scheda Bread Board del laboratorio inseriremo due resistenze che attraverso un lato si collegheranno al positivo (5 V) mentre l'altro capo si col-

Schema generale del funzionamento.

Montaggio degli ingressi sulla scheda Bread Board.

Collegamento dei pulsanti per il montaggio degli ingressi.

legherà alle molle 13 e 15. Le molle 14 e 16 sono unite mediante dei fili al negativo (0). Nella figura potete vedere il montaggio che stiamo spiegando.

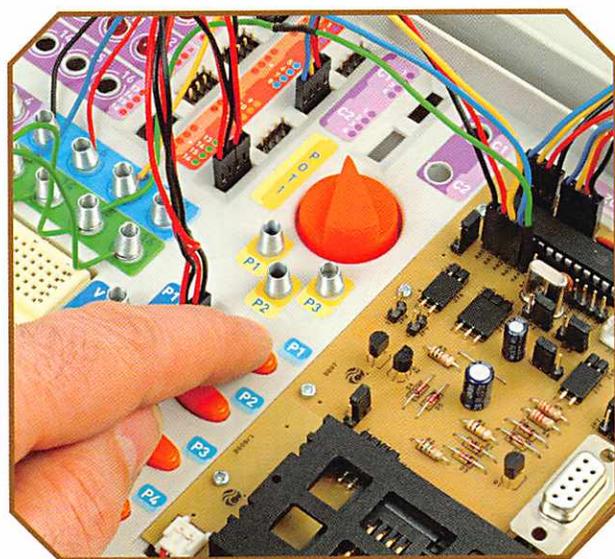
Ora dobbiamo montare i pulsanti al montaggio precedente, a questo scopo utilizzeremo due cavetti a due fili per unire i connettori corrispondenti alle molle di collegamento verdi con i connettori di P1 e P2. Il pulsante P1 si collegherà alle molle 13 e 14 e il pulsante P2 alle molle 15 e 16, come potete vedere nell'immagine in alto. Per portare gli ingressi al PIC è sufficiente unire la molla 13 con RA3, che corrisponde a P1 e la molla 15 a RA4, che corrisponde a P2. Per effettuare questo collegamento utilizzeremo un filo diretto dalla molla 15 al pin RA4 e uno dei fili di un connettore da quattro per collegare RA3 alla molla 13. Gli altri tre fili del cavetto di collegamento verranno portati alle molle azzurre 5, 6 e 7, in modo

Collegamento degli ingressi sul PIC.

Portiamo all'LCD i segnali di E, RS e R/W.

Colleghiamo le uscite del PIC all'LCD.

che la molla 5 sia collegata con RA0, la 6 con RA1 e la 7 con RA2.

Per terminare il montaggio degli ingressi dobbiamo collegare il connettore corrispondente alle molle azzurre con il display LCD. Mediante un cavetto uniamo i due connettori, in modo che la molla 8 corrisponda al pin situato più a sinistra dei connettori del display LCD del pannello superiore.

Montaggio delle uscite

Per eseguire il montaggio delle uscite è sufficiente collegare gli 8 pin della porta B con le 8 linee dei dati dell'LCD. Realizzeremo i collegamenti con due cavetti da quattro fili, tenendo presente che non possiamo incrociare i cavetti, perché se succedesse, i dati inviati al display sarebbero sbagliati. Nella figura potete vedere come è stato realizzato questo collegamento.

Premiamo P1 per scrivere un carattere.

Prova di funzionamento

Questo esercizio è uno dei più impegnativi da provare sul laboratorio. Acquisire la scioltezza necessaria per scrivere un messaggio correttamente, all'inizio richiederà un po' di allenamento, dato che si potrebbe incontrare qualche difficoltà a utilizzare i pulsanti.

Quando diamo alimentazione al laboratorio, agli ingressi RA4 e RA3 arrivano 5 V, dato che non sono premuti i pulsanti P1 e P2, il programma quindi si trova in modo programmazione, mostrando i caratteri in sequenza. I caratteri vengono visualizzati per il tempo definito nel programma. Per selezionare il primo carattere del messaggio dovremo premere P1 in modo che arrivi uno 0 sull'ingresso RA3. Quando il pulsante ritorna nella sua posizione originale il programma visualizza nuovamente tutti i caratteri sequenzialmente. In pratica dovremo premere P1 tante volte quante sono i caratteri contenuti nel messaggio.

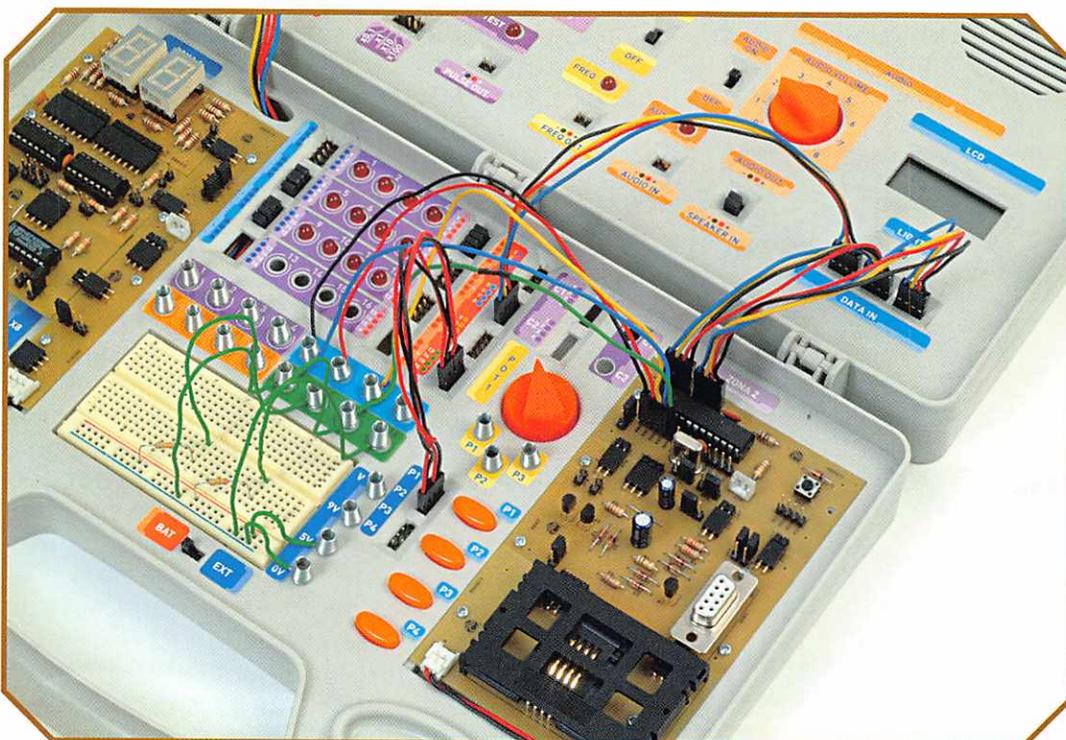
Quando il messaggio sarà completato lo potremo riprodurre premendo P2. Con questa azione arriverà uno 0 sull'ingresso RA4, quindi il PIC entrerà in modo riproduzione e si potrà vedere il messaggio sul display LCD

La riproduzione del messaggio si ottiene azionando P2.

Riproduciamo il messaggio sul display LCD.

per il tempo programmato. Nell'immagine potete vedere come viene visualizzato un messaggio temporaneo, precedentemente scritto.

Conclusioni

Abbiamo risolto questa applicazione in modo completo, ma ci sono ancora alcuni aspetti riguardanti il funzionamento del display LCD che non sono ancora stati evidenziati, ad esempio cosa succede quando si scrive un messaggio di più di 8 caratteri, che cosa succede con la memoria EEPROM quando si scollega l'alimentazione dal laboratorio, che risultato otteniamo dalle diverse possibili combinazioni degli ingressi o come agiscono sull'LCD i segnali E, RS e R/W. Provate, esercitatevi e modificate i programmi se lo riterrete necessario, dato che tutto questo vi aiuterà ad affrontare nuove applicazioni con maggior confidenza ed esperienza.

Aspetto generale del laboratorio.

Esercizio: generatore di messaggi (II)

Continuiamo l'applicazione del generatore di messaggi terminando lo sviluppo del codice ed eseguendo la compilazione e simulazione dell'esercizio. Sono già state definite le subroutine da utilizzare, l'intestazione del programma e la configurazione dei dispositivi del PIC.

Programma principale

Per continuare con il programma e affrontare la risoluzione dell'esercizio, dovremo avvalerci dell'aiuto dell'organigramma. La prima condizione prevista dopo la configurazione dei dispositivi, è lo stato dell'ingresso RA4. In funzione del valore RA4 selezioneremo il modo programmazione o riproduzione del messaggio. Il ciclo principale del programma – etichetta Loop – inizierà leggendo il valore dell'ingresso e prevedendo un salto condizionale in funzione del suo valore.

Modo programmazione

Nel caso in cui RA4=1 entreremo nel modo programmazione, quindi dovremo utilizzare una nuova variabile Temporale_1 per scrivere il primo carattere da visualizzare e leggere il valore dell'ingresso RA3 per sapere se il carattere visualizzato lo si deve memorizzare.

Se RA3=0 salteremo all'etichetta Scrivere, a partire dalla quale svilupperemo il codice necessario per scrivere il carattere nella memoria EEPROM. Se RA3=1 vedremo sul display i caratteri uno alla volta con un intervallo di 500 ms. Per fare questo posizioneremo prima il cursore nella posizione determinata dalla variabile Pos_Curs, visualizzeremo il carattere contenuto nella variabile Temporale_1, dopodiché richiameremo la routine di temporizzazione inizializzandola con il fattore 10 per ottenere 500 ms. Incrementeremo in seguito la variabile Temporale_1 per visualizzare il carattere successivo e, in funzione del fatto che sia o meno l'ultimo, ripeteremo il ciclo dal punto in cui assegniamo alla variabile il primo carattere o da quello in cui leggiamo lo stato di RA3. Nella figura in basso è riportato il codice corrispondente al blocco software che abbiamo appena visto.

Se RA3=0 scriveremo il carattere visualizzato. In questo caso, quindi, salteremo al codice localizzato a partire dall'etichetta Scrivere. La prima cosa che faremo sarà richiamare la routine per eliminare i rimbalzi, dopo ridurremo la variabile Temporale_1 per rimanere con il carattere visualizzato e, successivamente, lo scriveremo nella memoria EEPROM. Fatto questo, dobbiamo incrementare la posizione della memoria RAM per prepararci a scrivere un nuovo carattere, pulendola (ovvero scrivendo uno 0) per evitare possibili errori. Dobbiamo anche incrementare la posizione del cursore dell'LCD per presentare nuovamente i caratteri da scegliere.

Infine attenderemo che RA3 ritorni nuova-

```
messaggi - Blocco note
File Modifica Formato Visualizza ?
;***********************************************************************
;
Inizio          clrf    PORTB           ;Cancella i latch di uscita
                bsf     STATUS,RP0      ;Seleziona banco 1
                clrf    TRISB           ;Porta B si configura come uscita
                movlw   b'00011000'
                movwf   TRISA           ;RA0-RA2 uscite, RA3-RA4 ingressi
                movlw   b'00000110'
                movwf   ADCON1          ;Configuriamo la porta A come I/O digitali
                movlw   b'00000111'
                movwf   OPTION_REG      ;Prescaler da 256 per il TMR0
                bcf     STATUS,RP0      ;Seleziona banco 0
                movlw   0x80
                movwf   Pos_Curs        ;Posizione iniziale del cursore
                call    LCD_INI         ;Sequenza di inizio del LCD
                movlw   b'00000001'
                call    LCD_REG         ;Cancella LCD e Home
                movlw   b'00001110'
                call    LCD_REG         ;Invia istruzione: LCD ON, cursore ON e blink OFF
                bsf     STATUS,RP1      ;Passiamo al banco 2
                clrf    EEADR           ;Indirizzo iniziale della EEPROM
```

Configurazione dei dispositivi.

```
messaggi - Blocco note
File Modifica Formato Visualizza ?

Loop            bcf     STATUS,RP1      ;Passiamo al banco 0
                btfss   PORTA,4         ;Modo programmazione?
                goto    Riprodurre      ;No, modo riproduzione

Programmare     movlw   0x20
                movwf   Temporale_1     ;Primo carattere ASCII
Programmare_1   btfss   PORTA,3         ;Scrivi il byte nella EEPROM?
                goto    Scrivere        ;Sì
                movf    Pos_Curs,W
                call    LCD_REG         ;Posiziona il cursore
                movf    Temporale_1,W
                call    LCD_DATO        ;Visualizza il carattere
                movlw   .10
                movwf   Delay_Cont
                call    Delay_var       ;Temporizza 0.5 secondi
                incf    Temporale_1,F   ;Carattere ASCII successivo
                btfss   Temporale_1,7   ;È l'ultimo carattere ASCII?
                goto    Programmare_1   ;No
                goto    Programmare     ;Sì, iniziare dal primo
```

Codice per visualizzare i caratteri da scegliere.

```
messaggi - Blocco note
File  Modifica  Formato  Visualizza  ?

Scrivere        call    Delay_20_ms         ;Elimina rimbalzi
                decf    Temporale_1,W
                bsf     STATUS,RP1          ;Passiamo al banco 2
                movwf   EEDATA              ;Dato da scrivere nella EEPROM
                call    EE_Write            ;Scrivi il carattere
                bsf     STATUS,RP1          ;Passiamo al banco 2
                incf    EEADR,F             ;Indirizzo successivo della EEPROM
                clrf    EEDATA
                call    EE_Write            ;Scrivi 0x00
                incf    Pos_Curs,F          ;Posizione successiva del cursore
Scrivere_1      clrwdt                      ;Aggiorna il WDT
                btfss   PORTA,3             ;Attendi che RA3 ritorni a "1"
                goto    Scrivere_1
                call    Delay_20_ms         ;Eliminare rimbalzi
                goto    Loop                ;Carattere successivo del messaggio
```

Codice per salvare il carattere nella memoria.

mente a valore 1, istante in cui richiameremo la routine per eliminare i rimbalzi e torneremo all'inizio del programma all'etichetta Loop.

Modo riproduzione

Quando RA4=0 il PIC visualizza il messaggio scritto nella memoria EEPROM. I passaggi che seguiremo per sviluppare questo codice sono molto semplici. Per prima cosa prepariamo il display, lo cancelliamo e lo configuriamo. Dopo ci posizioneremo al primo indirizzo della memoria RAM e inizieremo un ciclo. All'interno del ciclo leggeremo il primo carattere della memoria e verificheremo che non sia l'ultimo; nel caso fosse l'ultimo salteremo a un'etichetta da dove richiameremo una routine di temporizzazione, inizializzandola con una variabile di valore 40 in modo che temporizzi due secondi. Se non è l'ultimo visualizzeremo il carattere, incrementeremo la posizione della memoria per andare al carattere successivo e ripeteremo il ciclo.

```
messaggi - Blocco note
File  Modifica  Formato  Visualizza  ?

Riprodurre      movlw   b'00000001'
                call    LCD_REG             ;Cancella LCD e Home
                movlw   b'00001100'
                call    LCD_REG             ;LCD = ON, cursore = OFF
                bsf     STATUS,RP1          ;Passiamo al banco 2
                clrf    EEADR               ;Indirizzo iniziale della EEPROM

Riprodurre_0    clrwdt                      ;Aggionamento del WDT
                call    EE_Read             ;Legge il carattere della EEPROM
                bsf     STATUS,RP1          ;Passiamo al banco 2
                movf    EEDATA,F
                bcf     STATUS,RP1          ;Passiamo al banco 0
                btfsc   STATUS,Z            ;E l'ultimo messaggio? (0x00)
                goto    Riprodurre_1        ;Si
                bsf     STATUS,RP1          ;Passiamo al banco 2
                movf    EEDATA,W
                bcf     STATUS,RP1          ;Passiamo al banco 0
                call    LCD_DATO            ;Visualizza il carattere
                bsf     STATUS,RP1          ;Passiamo al banco 2
                incf    EEADR,F             ;Carattere successivo
                goto    Riprodurre_0

Riprodurre_1    movlw   .40
                movwf   Delay_Cont
                call    Delay_var           ;Mantiene il messaggio per 2 secondi
```

Codice per riprodurre il messaggio.

Lampeggìo

Dopo aver riprodotto o visualizzato il messaggio per il tempo prefissato (2 s) il display inizia una sequenza di lampeggìo. Dovremo definire una nuova variabile Temporale_2 che svolga la funzione di contatore del lampeggìo. Ad esempio, possiamo inizializzare la variabile con un valore pari a 6, in modo che vengano

Ingresso	Stato	Descrizione
RA4	0	Riproduzione
	1	Programmazione
RA3	0	Visualizza carattere
	1	Scrivi carattere

Opzioni del programma in funzione degli ingressi.

```
Build Results
Building MESSAGGI.HEX...

Compiling MESSAGGI.ASM:
Command line: "C:\PROGRA~1\MPLAB\MPASMWIN.EXE /p16F870 /q C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM"
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 76 : Register in operand not in bank 0.  Ensure that b
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 77 : Register in operand not in bank 0.  Ensure that b
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 79 : Register in operand not in bank 0.  Ensure that b
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 81 : Register in operand not in bank 0.  Ensure that b
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 82 : Register in operand not in bank 0.  Ensure that b
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 96 : Register in operand not in bank 0.  Ensure that b
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 97 : Register in operand not in bank 0.  Ensure that b
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 106 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 108 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 110 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 112 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 122 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 147 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 150 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 151 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 165 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 170 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 175 : Register in operand not in bank 0.  Ensure that
Message[302] C:\PROGRA~1\MPLAB\PROGETTI\MESSAGGI.ASM 179 : Register in operand not in bank 0.  Ensure that

Build completed successfully.
```

Risultato della compilazione.

eseguiti 6 lampeggii. Accenderemo il display e attenderemo 250 ms, utilizzando la routine di temporizzazione, e spegneremo il display per un tempo pari al precedente. Fatto questo, ripeteremo il ciclo tante volte quante indicate nella variabile Temporale_2.

Terminato il lampeggio attenderemo un secondo e ripeteremo il programma dall'inizio, ovvero salteremo all'etichetta Loop.

Compilazione

Abbiamo terminato lo sviluppo del codice e dobbiamo compilarlo alla ricerca di eventuali errori.

Il programma completo potete trovarlo sul secondo CD allegato all'opera, all'interno della cartella LCD, con il nome "messaggio.asm". Creiamo un nuovo progetto su MPLAB cui allegheremo il nostro codice, lo visualizzeremo a video ed eseguiremo la compilazione. Il risultato di quest'ultima possiamo vederlo nell'immagine della figura in alto.

Simulazione

Per simulare l'applicazione apriamo le finestre abituali, quella dei registri delle funzioni speciali e una che contenga i registri che risultano essere più interessanti. Nella figura in basso potete vedere le variabili che abbiamo considerato come più importanti e il modo di visualizzarle (ASCII, decimale o binario). Inizieremo la simulazione passo a passo e, come

```
messaggi - Blocco note
File  Modifica  Formato  Visualizza  ?

;*************************** SEQUENZA DI LAMPEGGIO ****************************
Blink           movlw   .6
                movwf   Temporale_2         ;Inizializza il contatore di lampeggi
Blink_1         movlw   b'00001100'
                call    LCD_REG             ;LCD a ON
                movlw   .5
                movwf   Delay_Cont
                call    Delay_var           ;Temporizza 0.25 secondi
                movlw   b'00001000'
                call    LCD_REG             ;LCD a OFF
                movlw   .5
                movwf   Delay_Cont
                call    Delay_var           ;Temporizza 0.25 secondi
                decfsz  Temporale_2,F
                goto    Blink_1             ;Ripete il lampeggio
                movlw   .20
                movwf   Delay_Cont
                call    Delay_var           ;Temporizza 1 secondo
                goto    Loop

                end                         ;Fine del programma sorgente
```

Codice della sequenza per il lampeggio.

Watch_1

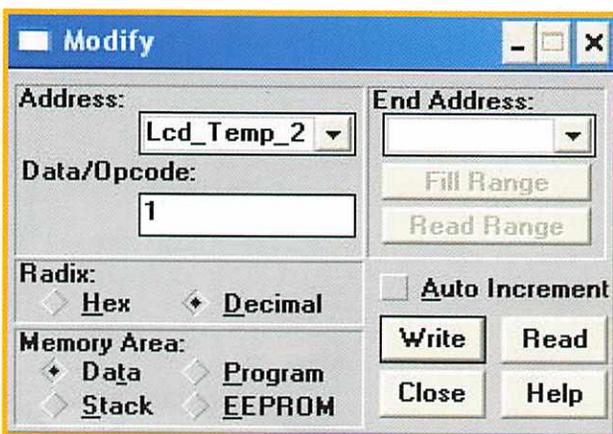
Address	Symbol	Value
06	PORTB	A'.'
05	PORTA	B'00000000'
22	Temporale_1	B'00000000'
23	Temporale_2	D'0 '
24	Pos_Curs	B'00000000'
21	Delay_Cont	D'0 '

Registri più importanti visualizzati in una finestra indipendente.

per gli esercizi precedenti, il simulatore entra nella libreria "Lcd_cxx.inc" per eseguire le routine del display LCD. Ricordate che per uscire da queste routine dobbiamo forzare le variabili Lcd_Temp_2 e Lcd_Temp_1 utilizzando la finestra Modify. Fatto questo e terminata l'inizializzazione del display, la simulazione continua all'interno del codice del programma principale configurando i dispositivi e il display LCD. La simulazione arriva al punto in cui si verifica lo stato dell'ingresso RA4. Dato che non abbiamo forzato il suo valore è chiaro che l'ingresso sarà a zero, quindi ci troviamo nel modo riproduzione. Siccome non c'è alcun messaggio registrato nella memoria, il programma presenta il contenuto della memoria stessa ma non trova la fine del messaggio.

MPLAB durante la simulazione.

Abbiamo quindi trovato un errore, infatti, dato che l'applicazione alla partenza entra nel modo riproduzione, non trovando la fine del messaggio continua a riprodurre costantemente il contenuto della memoria EEPROM. Possiamo risolvere questo problema in due modi, il primo consisterebbe nel provocare un interrupt ogni volta che cambia lo stato dell'ingresso RA4 (in questo modo usciremmo dal ciclo), e il secondo, più semplice, sarebbe quello di cancellare il contenuto del primo indirizzo della RAM al momento di configurare i dispositivi. Nell'immagine possiamo vedere come risolvere questo errore.

Come avete potuto verificare, mediante la simulazione è possibile rilevare degli errori di funzionamento che in altro modo non potremmo evidenziare. Modificate il programma utilizzando l'editor di MPLAB, salvatelo e compilate nuovamente. Ripetete la simulazione e verificate che ora avvenga in modo adeguato. Per eseguire una simulazione completa sarà necessario utilizzare il simulatore di stimoli asincroni e attivare gli ingressi, in questo modo sarà possibile analizzare le diverse combinazioni verificando il funzionamento completo dell'applicazione.

Forziamo i valori delle variabili.

c:\progra-1\mplab\progetti\messaggi.asm

```
              movlw   b'00000111'
              movwf   OPTION_REG     ;Prescaler da 256 per il TMR0
              bcf     STATUS,RP0     ;Seleziona banco 0
              movlw   0x80
              movwf   Pos_Curs       ;Posizione iniziale del cursore
              call    LCD_INI        ;Sequenza di inizio del LCD
              movlw   b'00000001'
              call    LCD_REG        ;Cancella LCD e Home
              movlw   b'00001110'
              call    LCD_REG        ;Invia istruzione: LCD ON, cursore ON e blink OFF
              bsf     STATUS,RP1     ;Passiamo al banco 2
              clrf    EEADR          ;Indirizzo iniziale della EEPROM
              movlw   b'00000000'
              movwf   EEDATA         ;Dato da scrivere nella memoria EEPROM (resetta)
              call    EE_Write       ;Scrive il carattere
              clrf    EEADR          ;Indirizzo iniziale della EEPROM

Loop          bcf     STATUS,RP1     ;Passiamo al banco 0
              btfss   PORTA,4        ;Modo programmazione?
              goto    Riprodurre     ;No, modo riproduzione

Programmare   movlw   0x20
              movwf   Temporale_1    ;Primo carattere ASCII
Programmare_1 btfss   PORTA,3        ;Scrivi il byte nella EEPROM?
              goto    Scrivere       ;Sì
              movf    Pos_Curs,W
              call    LCD_REG        ;Posiziona il cursore
```

Modifica per correggere l'errore.